DOCUMENT RESUME

ED 427 737 IR 019 288

AUTHOR Svensson, Lars; Ekenstam, Tobias
TITLE Web Education for Those Who Don't Know How but Want To, and for Those Who Know How but Don't Want To.
PUB DATE 1998-11-00
NOTE 7p.; In: WebNet 98 World Conference of the WWW, Internet and Intranet Proceedings (3rd, Orlando, FL, November 7-12, 1998); see IR 019 231. Figures may not reproduce clearly.
PUB TYPE Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; Computer Literacy; Computer Mediated Communication; *Computer System Design; Course Content; *Distance Education; *Instructional Materials; Learning Activities; Material Development; Models; Student Projects; Teacher Attitudes; Teaching Methods; *World Wide Web
IDENTIFIERS Computer Use; *Web Sites

ABSTRACT

One obvious reason that teachers hesitate to create Web sites for their distance courses is the lack of computer/technology skills. Other teachers who know how to create a well-structured course site are put off by the fact that it takes a lot of time to maintain it. Thus there is a risk that the course site will be ill-structured and poorly maintained. This paper presents a system aimed at overcoming both of these obstacles. The system DisCo (Distance Courses) is designed to help teachers with their basic needs for using the Web to publish course material and to communicate with their students. Course related activities of the DisCo system are divided into four basic categories: Information, Course Material, Communication, and Projects, each containing several subcategories. In DisCo, all course site maintenance is done via the Internet. Evaluations show that students appreciate the system, although it should be noted that all students have had a 2-week course in basic computer use prior to attending their first course. Contains 11 references. Contains 11 references. (AEF)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 427 737

Web Education for those who don't know how but want to, and for those who know how but don't want to.

PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

G.H. Marks

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

Lars Svensson
Department of Economy and Computer Science
University of Trollhättan Uddevalla, Sweden
lars@udd.htu.se

Tobias Ekenstam
Department of Economy and Computer Science
University of Trollhättan Uddevalla, Sweden
ekenstam@udd.htu.se

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

Abstract: The World Wide Web is a great medium for distance education, but it can also be tough to master. To create and maintain a good hypertext-system for a distance-course can be a truly time-consuming task that requires a lot of computer know-how. Many teachers lack the knowledge and/or is reluctant to spend the time necessary. This paper presents a system designed as a framework for using the www in distance education. The system (DisCo – Distance Courses) aims at helping teachers with their basic needs for using the web to publish course material and to communicate with their students, thereby inviting both those who do not know how to as well as those who do not want to.

1 Introduction

Internet technology provides a number of properties that would be tempting for most teachers to incorporate in their courses. Especially this would apply to distance education, an area where the Internet is becoming the dominating media. The possibility to arrange course material in a hypermedia structure, provide students with interactive educational software and communicate with students via email and online-conferences are some of the things that could be facilitated. Using the Internet to its full capacity is hardly an issue for most teachers. Instead it is a question of more basic needs like the wish to publish text-based material and tutor students. Already these simple needs can prove to be difficult enough to make a teacher hesitant in setting up a web site for his or hers distance-course. One obvious reason is the lack of computer/technology skills. Others who know how to create a well-structured course-site are put off by the fact that it takes a lot of time to maintain it. There is therefore a risk that a course site will be ill structured and poorly maintained. In this paper we present a system, aimed at overcoming both these obstacles.
The ideas originally evolved from the author's own time-consuming efforts to create maintain and administer web-sites for campus-courses. Working with three or four 'DisCo-courses' simultaneously surfaced the problems that Ljungberg and Sorensen calls Interaction Overload [Ljungberg & Sorensen 1998]. Several experiences served as triggers for wanting to create a system for web education, some of them listed below.

1. Adding new documents to the site involved re-coding existing web-pages. (When more than one teacher is involved in the course, there are problems updating the same file simultaneously)
2. When giving several courses at the same time it is hard to separate incoming email. (Students tend to forget to tell which course they are taking, which makes it hard to answer questions like "When is the assignment due?" or "Can I have the study Guide for chapter two in Mac-word format, please?")
3. Problems with submission of digital documents. (Sixty students sending email with an attached file called exercise1.doc)

At the same time two distance-education projects where initiated at the University of Trollhättan Uddevalla. Both these projects would involve teachers that had little or no experience of web authoring. This fact was the final trigger, and we started working on the system in January 1997. A first prototype was taken in to use in March the same year. In the first list of system requirements we attempted to address both the issues of non-internet skilled teachers as well as the time saving aspects of maintenance. The system should be able to:

IR019288

BEST COPY AVAILABLE

- Support the possibility for teachers to publish documents, created with software-packages of their own choice.
- Provide a flexible, user-friendly hypertext structure that was easy to navigate.
- Ensure that hand-ins and email from students were well documented
- A small set of general actions are used to maintain the course-site
- No programming skills are required for the course-provider.
- All functions in the system can be reached over the Internet through a standard web browser, eliminating the need for the teacher to be present at school.

The system has been used as a distance-component in eight campus-courses since the start in 1997. In January 1998 the SYDUB-project started, where DisCo was tried for the first time in a distance educational setting. Ever since the start, both students and colleagues have been helpful in evaluating and improving the system This evolutionary approach [Dahlbom & Mathiassen 1993] to system development has implied many changes to interface and functionality. For instance, the original set of images used for navigating the system was at one time replaced, due to complaints from students using slow modem access.

2 Student Interface

Interaction is a buzzword with no strict definition [Jensen 1997]. But attempts to define the word from an educational point of view have been done by [Moore 1993] who distinguishes between learner-content, learner-instructor and learner-learner interaction. [Poulsen 1995] presents four pedagogical techniques for computer-mediated communication that focuses on the number of interactors (One-alone, one-to-one, one-to-many, and many-to-many). These categorisations can of course be useful in discussing the different kinds of interactions supported by the DisCo-system. However neither of them guided us when deciding on how to divide all course-related activities into the categories that were to form the basis of the hypertext structure of a course-site. Instead we took a more practical starting point, , focusing on the initiator of the interaction. We choose to use four main categories; Information, Course Material, Communication and Projects, each containing several subcategories. Two of the categories (Information and Course Material) supports interaction initiated by the teacher, (learner-instructor /one-to-many). Communication collects different ways for a student to interact (learner-instructor, learner-learner/ one-to-one, many-to-many). Finally, the Project category gives groups of students possibility to a more private interaction. Neither Moore's nor Paulsen's definitions of interaction could be strictly applied to the categories and sub-categories of DisCo, this is to a large extent due to the openness of the system. Most categories can be used according to the preferences of the course provider. Below is a short description and the intended use of each category presented. (See [Table 1] for description of all subcategories)

Information: The reason for splitting the teacher-initiated interaction in two categories was the idea that the information category should contain static files that could be present on the site when the course started, with low probability of being changed during the course. (The Course Material category was intended to be more dynamic in nature.) For instance a description of the course, its content, goals and methods for examination can be found here along with a presentation of involved teachers and help for students on how to configure their browser

Course Material: Most subcategories consist of dynamically generated listings with different kinds of content. Each item in such a list is a hyper-linked file-name, supplemented with a text abstract describing the content of the file. At the moment there are separate listings for material oriented toward study guidance, description of assignments and collection of external web-resources. A slightly different kind of listing generates an on-line self-correcting quiz, to be used by students in order to test themselves [Carbone & Schendzielorz 1997]

Communication: In this category we collected several techniques for student initiated interaction. First of all the students add their name and email address to a course-mailing list. To make students responsible for providing their correct address lifts a heavy burden from the shoulders of central administration, especially since many students have two or more different email addresses.
The next possibility to interact is to send email to the teacher(s) of the course. Instructing students to use the

DisCo e-mail form instead of their regular mail software has the advantage that the subject line can be automatically generated, a fact that allows the teacher to effectively filter his incoming mails into a course-specific mailbox. For students to submit their assignments as digital documents instead of handing in paper copies can in many situations be preferred. But as described earlier, the administration of these documents can be very time-consuming. In DisCo such digital hand-ins are uploaded to the web-server instead of being attached to email, (or even worse – handed over on a floppy disk).. The dialogue for submitting a hand-in, forces the students to write a rich documentation on the identity of the assignment (name of exercise, group-ID, name of authors, software-version etc). Thanks to this, there is no need for creating complex rules on how to name the files (which are simply renamed on submission) . Finally we have incorporated a threaded Debate Board, primarily intended for communication amongst students. Here they can post messages, ask questions and react to entries made by others. The teacher administrates the board.

Projects: Since more and more activities in both regular and distance education are done in groups, there is a need to support the collaboration of such groups even though they might be geographically dispersed, or just having difficulties to find the time to meat – face to face. This category is in a way a DisCo in miniature, a primitive groupware-tool where group members can upload files and engage in debate in their own password-protected area.

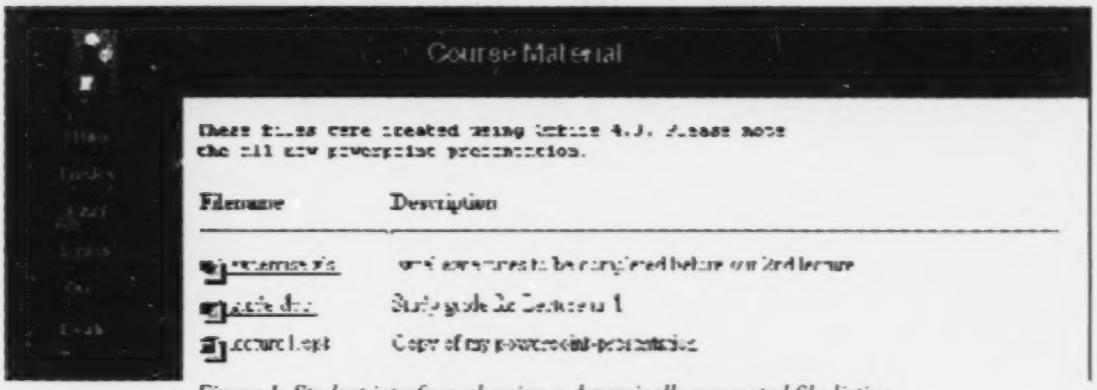

Figure 1: Student interface, showing a dynamically generated file-listing.

Navigating this two-level interface is done by buttons placed in frames at the top and to the left of the screen. Clicking on first level buttons updates the left frame, showing accurate sub-category buttons. Colours are used to indicate which buttons are pressed, thereby indicating the nature of the material in the content-area of the screen. Students can also choose to access the system through a non-frames, non-java script interface

Information		Course Material		Communication		Projects	
News	Bulletin board with latest information	**Files**	List with study-guides, lecture-notes, exercises etc	**Sign Up!**	Adding your name to the mailing list	**Files**	Adding and deleting share documents
Content	Course description	**Tasks**	Detailed instructions for assignments	**Email**	Send an email to the teacher(s)	**Links**	Group collection of external URLs
Teacher	Presentation and addresses	**FAQ**	Frequently Asked Questions	**Hand-In**	Submit a file for an assignments	**Email**	Administer the group-mailinglist
Schedule	Dates and reedings for lectures assign-ments etc.	**Links**	Interesting course related web-sites	**Debate**	Threaded debate-board	**Debate**	Group-debateboard
Help	Users-Guide and Download sites for viewers	**Quiz**	Online self-correcting questionnaire	**Evaluate**	Form for evaluat-ing the system	**Admin**	Change password and allowed fileformats
		Exam	Results from exams				

Table 1: All Categories and sub-categories of the DisCo- System

BEST COPY AVAILABLE

3 Maintaining a DisCo

When discussing the meaning of the term, Distance Education, the focus is often on students being separated from the school (in time and space). In DisCo, all course-site maintenance is being done via the Internet. A fact that adds another dimension to the concept of Distance Education; not even the teacher has to be present at school. This shift from student to teacher focus can be said to be the trademark or bearing concept of DisCo. Without teachers committing themselves to web education, little will be accomplished. The philosophy was to make a simple, yet powerful interface for the maintenance of the course site, minimising the need for computer skill and time.
The administrative part of DisCo is located to a password-protected area of the web-server, with a start-page containing links to maintenance-pages for all ongoing courses and some shared resources for all course-providers. I.e. a debate board where for instance suggestions of new features, reports on bugs and added functionality can be posted and discussed. This page also holds the possibility to create a new course. In doing so a small configuration file, containing names and email addresses for all teachers as well as format specifications for file-listings, is created. The teacher can alter these settings at any time. Furthermore the creation of a new course triggers a small perl-script that among other things is responsible for creating a predefined directory-structure on the web-server, to which all files used in the student interface are copied. This enforced standard allows the advanced author to use internal links from one html document to another.
In order to make maintenance easy, even for novice course-providers, we felt it was important to reduce the set of maintenance-actions to a minimum. There are basically two different techniques or actions that a teacher needs to master in order to provide a well-structured course to his students.

3.1 Edit text files

A number of the categories in DisCo contain static text/html documents. The technique to create or update this file is the same for all these documents. After choosing what file to update, the present content of the file is uploaded to a text-box where it can be edited by the teacher, and then re-submitted to the server. Since the text-box does not offer WYSIWYG-facilities, the result on how the file looks like viewed through a browser, is presented. If not satisfied, the teacher can repeat the process. If a richer layout and design is wished for, the more experienced author can choose either to hardcode html-tags into the text-boxes, or create complete html files in an editor of his/her own choice. If an external editor is used, the file is submitted to the server, given a file name matching the content of the file. (Of course images or other referred documents must be submitted separately.) Example of files that can be edited this way are the headers for the dynamically generated file-listings of the Course Material category, the static files of the Information category but also more dynamic documents like the FAQ, (Frequently Asked Questions), and the news-bulletin that serves as the default starting page for a course.

3.2 Show-Add-Delete

The other basic technique is used to maintain the content of the generated file-listings of the Course material category. For each of these, the interface has three buttons to serve this purpose. One to show the present content of the listing, one used to add a file (plus-sign), and one to remove a file from the list (minus-sign). In the add-file dialogue the teacher marks the file to be uploaded and writes a short text description of the content. If the format of the submitted file matches the specifications in the course configuration file, mentioned earlier, it will be included in the listing. Picture-files (GIF, JPG) are normally not formats that are specified to be listed, but can still be uploaded and referred to from within html documents on the listing.

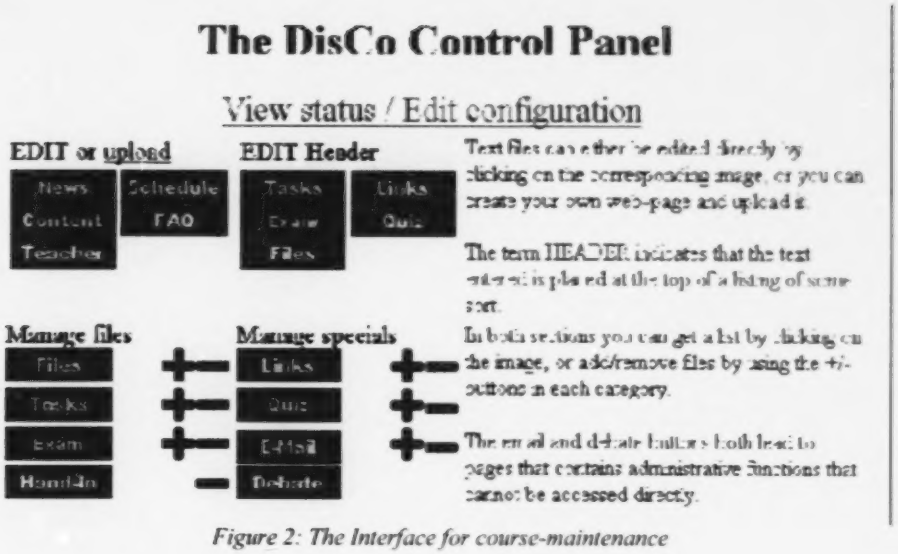

Figure 2: The Interface for course-maintenance

3.3 Push

An often discussed property of the WWW is that it is Pull-technology [Wired, March 1997]. I.e that a user is forced to actively find the information and retrieve it through activating a hyper-link. In contrast, the term Push-technology is used to describe situations where information [Wired March 1997] seeks its receiver without being called for. The easiest way to facilitate a Push technology that makes the student aware of new entries to the course-site is to use e-mail [Stegberg & Svensson 1997]. The dialogues of adding a file or updating a text document both contains the possibility to check an option that will generate emails to all persons that has registered their address on the mailing list of the course. The message consists of information on where the new file is available (with complete URL) and the short description-text on the content.

4 DisCo Tech

The overall philosophy for technical solutions in the DisCo-system has been Easy and Cheap. DisCo is currently hosted on a P133Mhz with 32mb memory running RedHat Linux 4.2 and NCSA HTTPd 1.5.2a. DisCo consists of a number of small cgi-programs written in Perl (and occasionally, C). There are also a number of static HTML-files, for example the navigation frames, help pages, etc. A new course is created by filling out a form with brief course information. The cgi-program connected to the form uses an empty template course that is copied to a new course. A regular expression search and replace enters the correct values in the static html files and ads the proper links so that the course becomes accessible. The removal of a course is done in the same manner. A search and replace removes the links and the created directories are deleted. The majority of the pages the student sees are generated on the fly. By using the referrer-variable, as well as command line arguments, the same script can be used by all courses, which makes the system easy to maintain and update. The teacher uses the same scripts to view information, but also has access to the add/remove scripts. Adding files is done with html-forms using the multipart/form-data encoding type, which eliminates the need of additional ftp-software. The removal of files (or other resources such as links, student email-addresses, questions in the quiz) is done by simple point-and-click forms.

5 Closing remarks

We feel that most systems for web education are oriented towards a user-perspective. There are many impressive

BEST COPY AVAILABLE

examples facilitating multimedia content, query-databases and on-line applications. For example see [Ahanger & Little 1997], [Carbone & Schendzielorz 1997] or [Bodner et. al 1997]. We have shifted the focus and concentrated on the basic needs of the novice-author, still allowing more experienced web-authors to do their tricks. The student interface of DisCo is designed for usability and to give a satisfactory level of structure to all course related activities

The system requirements previously listed [see Introduction], are all items in favour of easy authoring and maintenance but there are of course also prices to be paid from these simplifications. For instance, when encouraging teachers to publish files created in non-html formats, the possibility of providing full-text-search is limited. It also restrains the author from being able to create internal links from a hot spot within one document to a target within another [Nielsen 1995].

Even though our main concern when designing DisCo has been the course providers, we do not feel that users (students) are suffering from this. Evaluations so far show that students appreciate the system, although it should be noted that all students have had a two-week course in basic-computer-use prior to attending their first course. Using the system in a true 'novice-setting' (or with non-university students) could give a different result

We have several ideas on how to further develop the DisCo system. One of the more interesting is to develop the course creation function towards customisation, hence the teacher should be able to choose what categories should be used in the course. We also want to facilitate customisation at the user-end of the system. Many students take two or more courses at the same time and could therefor benefit from designing his/hers own DisCo-interface.

6 References

[Ahangger & Little 1997] Ahanger, G and Little, T.D.C (1997) *Easy Ed: An Integration of Technologies for Multimedia Education* Proceedings of WebNet 97, AACE

[Bodner et al 1997] Bodner, R and Chignell, M and Tam, J (1997) *Website authoring using Dynamic Hypertext* Proceedings of WebNet 97, AACE

[Carbone & Schendzielorz 1997]Carbone, Angela and Schendzielorz, Peter (1997) *Developing and Integrating a Web-based Quiz into the Curriculum* Proceedings of WebNet 97, AACE

[Dahlbom & Mathiassen 1993] Dahlbom, Bo and Mathiassen, Lars (1993) *Computers in Context: Philosophy and Practice of Systems Design* – Cambridge: Blackwell

[Jensen 1997] Jensen, Jens F (1997) *'Interactivity' - Tracking a New Concept* Proceedings of WebNet 97, AACE

[Ljungberg & Sorensen 1998] Ljungberg, f and Sorensen, C (1998), *Are you "Pulling the plug" or "Pushing up the daises?.* Proocedings of HICSS 31

[Paulsen 1995] Paulsen, Morten F (1995) *The Online Report on Pedagogical Techniques for Computer-Mediated Communication.* <http://www.nki.no/~morten/cmcped.htm

[Moore 1993] Moore, Michel G (1993) *Three types of interaction. Distance Education: New Perspectives* edited by Harry, Keegan and John– New York: Routledgesp.19

[Nielsen 1995] Nielsen J (1995). *Multimedia and Hypertext: The Internet and beyond*

[Stegberg & Svensson 1997] Stegberg, T and Svensson, L (1997) *How to use the Internet, when you don't have to.* Proceedings of the IRIS 20. Department of Informatics, University of Oslo

[Wired, March 1997]The editors of Wired (1997) *PUSH! Kiss your browser goodbye: The radical future of media beyond the Web* – Wired, March 1997

URL to the DisCo-system: http://www.udd.htu.se/dl/test/frames.html

7 Acknowledgements

This work is partially financed by the SYDUB-project (financed by EU, HTU and local communities). Several persons have been extremely helpful in the process of developing DisCo. Special thanks to Leon Vermaas who designed the first user-interface and did a lot of tedious work on documenting the system. Many thanks to colleagues and friends in the Internet Project for valuable comments on the paper.